

# Esercitazione - DroNET

Reti di Elaboratori - 18/19

Corso di Laurea in Informatica

Università degli Studi di Roma “La Sapienza”

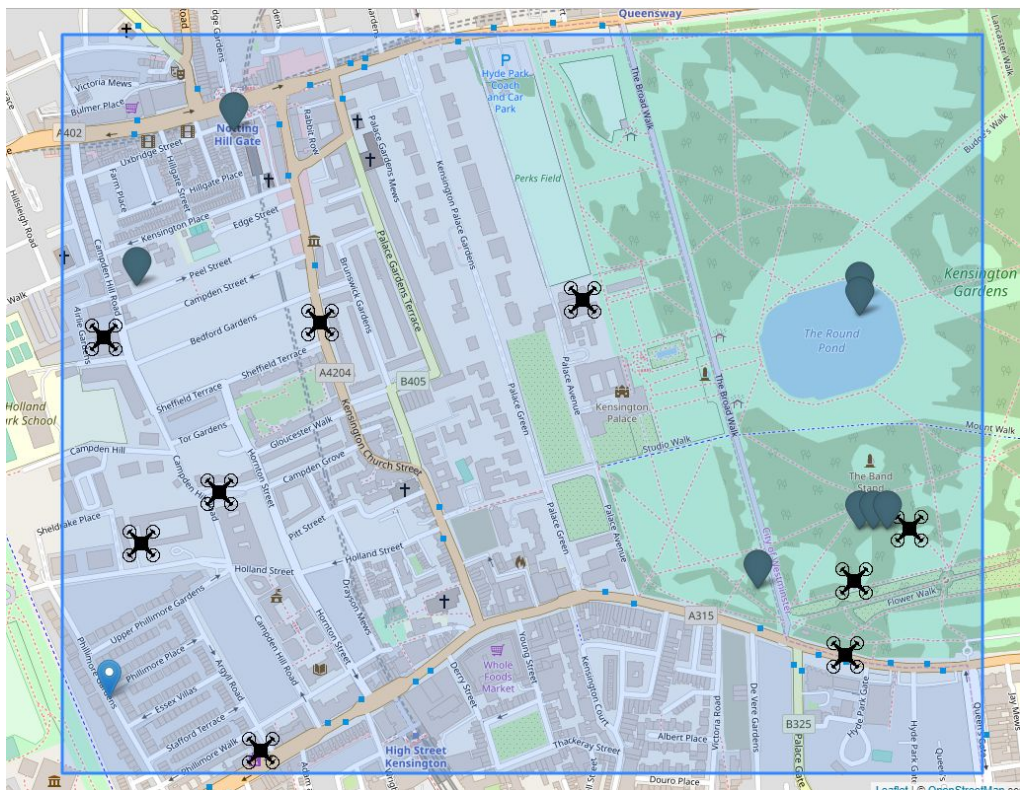


[bit.ly/drone-ms](https://bit.ly/drone-ms)

Mauro Piva - Andrea Coletta  
{coletta | piva}@di.uniroma1.it

# DroNET

- Mappa con punti di interesse
- Un drone per ogni utente
- Ogni drone può muoversi in 8 direzioni:
  - Nord - Sud
  - Est - Ovest
  - NordEst - NordOvest
  - SudEst - SudOvest
- A POI (Point-Of-Interest)



# Obiettivo finale

---

Creare un drone e visitare più punti possibile sulla mappa!



**Non far schiantare il drone**



# DroNET - Come funziona



Server

35.228.1.72



Clients

# DroNET - Server



Server  
35.228.1.72



Accetterà solamente  
richieste HTTP BEN  
FORMATTATE!

# DroNET - Client

---

- Il client dovrà permettere di:
  - Scaricare la mappa e i dati di gioco
  - Creare un nuovo drone
  - Elaborare la scoreboard
  - Pilotare/gestire il drone



# DroNET - Rules

---

- 5 livelli
- Python (preferibilmente) / Java
- Non è consentito:
  - Utilizzare librerie come **urllib2**, **requests**, **HTTPUrlConnection**, **HttpComponents**, etc...
- E' consentito invece l'utilizzo di:
  - **socket** (python - Java), **json**, **sys**

# DroNET - Details

---

- IP Server: 35.228.1.72
- Mappa online con i droni: <http://35.228.1.72/dronet-ms-core/v1/getmap>
  - Aggiungere “**droneUnique=identificativo\_drone**” per evidenziare il proprio drone

Trick:

- <http://35.228.1.72/swagger-ui.html#/>



# DroNET - Details



- Problemi di linea?
- `$ git clone https://github.com/mauropv/dronet-ms`
- `cd dronet-ms`
- `mvn clean install`
- `sudo java -jar dronet-ms-core/target/dronet-ms-core-1.0.0-SNAPSHOT.jar`
- Nuovo endpoint! (localhost)

# DroNET - client details

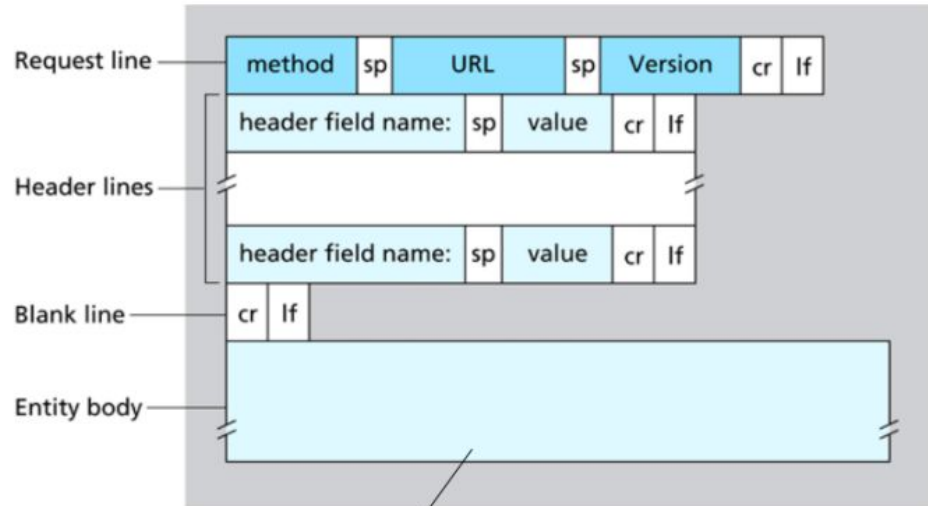
Come fare?

Scrivere le chiamate HTTP.

In particolare:

- GET
- POST
- PUT

Formato generale dei messaggi di richiesta HTTP:



Campo vuoto per il GET, utilizzato per il POST

## Remember:

- <http://35.228.1.72/dronet-ms-core/v1/getmap>
- Non persistent connection

# DroNET - Step 1

---

- Scaricare dati di gioco (0min)

**URL:** GET /dronet-ms-core/v1/mapStatus

**Descrizione:** Restituisce un oggetto JSON contenente una serie di informazioni sulla mappa

**Parametri: Header:** user-agent deve essere chat-client

**Body:** ?

**Response:** un oggetto JSON contenente le info sul gioco/mappa

**Obiettivo:** Stampare a schermo il **numero di droni** presenti, i loro **nomi**, il **numero di POI** e i dettagli della mappa (dimensioni e posizione della “base station”)

Remember:

- <http://35.228.1.72/dronet-ms-core/v1/getmap?droneUnique=uniqueID>

## DroNET - Step 2

---

- Creare un nuovo drone (20min)

**URL:** PUT /dronet-ms-core/v1/createDrone

**Descrizione:** Permette di creare un nuovo drone

**Parametri: Header:** user-agent deve essere drone-client

**Body:** droneName (scegliete voi il nome del drone)

**Response:** un oggetto JSON contenente uniqueId e secret. Dovrete riutilizzare questi parametri nelle successive comunicazioni.

**Obiettivo:** Creare un nuovo drone con il vostro nome e trovarlo sulla mappa

Remember:

- <http://35.228.1.72/dronet-ms-core/v1/getmap?droneUnique=uniqueID>

## DroNET - Step 3

- Dare comandi ad un drone (15min)

**URL:** POST /dronet-ms-core/v1/updateDroneStatus

**Descrizione:** Permette di aggiornare la direzione del drone.

**Parametri: Header:** user-agent deve essere drone-client

**Body:** deve contenere un json con chiavi “**direzione**”, “**idunivoco**” e “**secretkey**”, i valori devono avere tipo **string**.

“**direzione**”  $\in$  {“N”, “S”, “O”, “E”, “NE”, “NO”, “SE”, “SO”}

**Response:** un oggetto JSON contenente una serie di informazioni sul drone

**Obiettivo:** Muovere il drone prendendo in input la **direzione** da tastiera. (Attenti al content-type)

Remember:

- GET /dronet-ms-core/v1/mapStatus

## DroNET - Step 4

- Visualizzare una leaderboard (10min)

Utilizzare il JSON ottenuto dalla chiamata:

- GET /dronet-ms-core/v1/mapStatus

Aggiornare periodicamente la leaderboard (0.5 sec?)

**Obiettivo:** Visualizzare i droni che stanno vincendo.

Tips.

```
>import os
```

```
if os.name == 'nt': os.system('cls')
```

```
else: os.system('clear')
```

LEADERBOARD	
AzureAngel	1934 pts
CR90 Corvette	1905 pts
Death Star	1725 pts
Ghost	1676 pts
....	1574 pts
	735 pts
	467 pts
	156 pts

Remember:

- <http://35.228.1.72/dronet-ms-core/v1/getmap?droneUnique=uniqueID>

# DroNET - Step 5

## 1. Sparare (10min)

**URL:** PUT /dronet-ms-core/v1/createShot

**Descrizione:** Permette di creare un missile e spararlo.

**Parametri: Header:** user-agent deve essere drone-client

**Body:** deve contenere un json con chiavi “idunivoco” e “secretkey”, i valori devono avere tipo **string**.

**Response:** un oggetto JSON contenente una serie di informazioni sullo sparo.

**Obiettivo:** Creare uno sparo a seguito della pressione del tasto “space”. (lo sparo si muoverà nella direzione del drone)

Remember:

- <http://35.228.1.72/dronet-ms-core/v1/getmap?droneUnique=uniqueID>

# DroNET - Step X

## 1. Autopilot & Team Up & Play

Descrizione: -creare droni autonomi:

-creare team di più droni

Steps:

1-Autopilot

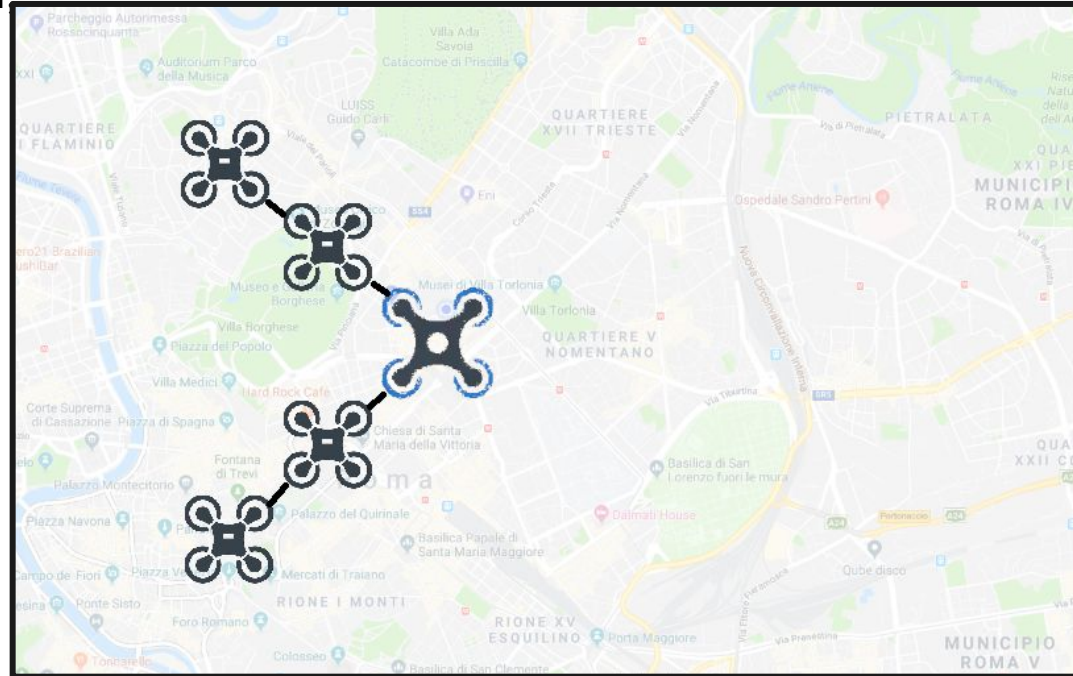
2-Creare più droni

2-Riunirli

3-Coordinarli

3.1- Seguire il leader

3.2 - Droni autonomi





# Pssw zip:



step\_1 --> no pssw

step\_2 --> questo\_codice\_serve\_per\_step\_2

step\_3 --> step\_3\_usa\_il\_seguinte\_codice

step\_4 --> dovreste\_usarlo\_per\_fare\_lo\_step5

step\_5 --> ultimo\_sforzo\_per\_finire

**Let's  
Play!**